

⑨ 日本国特許庁(JP)

⑩ 特許出願公開

⑫ 公開特許公報(A)

平2-100717

⑪ Int. Cl.<sup>3</sup>

識別記号

庁内整理番号

⑬ 公開 平成2年(1990)4月12日

G 06 F 7/52

3 2 0 D

7056-5B

審査請求 未請求 請求項の数 1 (全8頁)

⑭ 発明の名称 固定小数点除算方式

⑮ 特 願 昭63-252905

⑯ 出 願 昭63(1988)10月8日

⑰ 発 明 者 牧 田 明 久 東京都港区芝5丁目33番1号 日本電気株式会社内

⑱ 発 明 者 桜 井 博 山梨県甲府市丸の内1丁目17番14号 甲府日本電気株式会社内

⑲ 出 願 人 日本電気株式会社 東京都港区芝5丁目33番1号

⑳ 出 願 人 甲府日本電気株式会社 山梨県甲府市丸の内1丁目17番14号

㉑ 代 理 人 弁理士 芦 田 坦 外2名

## 明 細 書

## 1. 発明の名称

固定小数点除算方式

## 2. 特許請求の範囲

1) 正の整数除算を行う固定小数点除算方式において、

除数、被除数の有効桁の最上位桁を、除数、最初の剰余の最上位桁に併合せする  $2^n$  進併合せシフト ( $n \geq 1$ ) と、

該併合せされた除数を  $+1, -1, -2, -4, \dots, -2^{n-1}$  倍数発生回路と、

前記除数、被除数の有効桁より商の有効桁を事前に算出する有効桁算出回路と、

前記倍数発生回路の出力と併合せされた剰余を入力し、剰余から併合せされた除数の仮の部分商倍 ( $+1, +2, +3, \dots, (2^n - 1)$ ) された数をそれぞれ OSA トゥリーを用いて減算し、仮の剰余のキャリーデータとサムデータを得て、

仮の剰余の符号を求め、負とならない最大の仮の剰余および部分商を選択し、真の剰余と真の  $n$  ビットの部分商を得る選択回路と、

商が有効桁となるまで部分商を  $n$  ビットづつくり返し求め、商の最下位桁が求まった後、キャリーデータとサムデータを加算し、剰余とする加算回路と、

剰余を被除数が併合せした量だけ右シフトさせるシフトとを有することを特徴とする固定小数点除算方式。

## 3. 発明の詳細な説明

〔産業上の利用分野〕

本発明は固定小数点除算方式に関し、特に正の整数の除算で、商の有効桁を事前に求め、OSAを用いて部分商を  $n$  ビット単位で有効桁だけ求める固定小数点除算方式に関する。

〔従来の技術〕

正の整数除算を行う固定小数点除算方式は従来、商を1ビットづつ求めていた。この従来例

を第4図に示す。被除数はレジスタ16, 17へ上位, 下位と分けて格納され, 除数はレジスタ18へ格納される。レジスタ16, 17は連結してシフト動作する。レジスタ16に格納された部分被除数とレジスタ18より補出力される除数は $div=1$ としてアダー19で減算が実行される。 $cout=1$ ならば減算結果は正で, 部分商は1となり, レジスタ2の下位1ビットに出力され, 減算結果がレジスタ1へ入力される。 $cout=0$ ならば, 減算結果は負で, 部分商は0となり, レジスタ17の下位1ビットに出力され, レジスタ16は前の値を保持する。減算を実行するごとにレジスタ17のデータは1ビットづつ左シフトされレジスタ16の下位1ビットへ入力される。

以上の動作をレジスタ17のデータがシフトし終わるまでくり返すことによりレジスタ16のデータが剰余にレジスタ17のデータが商になる。

#### 以下余日

( $+1, +2, +3, \dots, +(2^n - 1)$ )された数をそれぞれCBAトリユーを用いて減算し, 仮の剰余のキャリーデータとサムデータを得て, 仮の剰余の符号を求め, 負とならない最大の仮の剰余および部分商を選択し, 真の剰余と真の $n$ ビットの部分商を得る選択回路と, 商が有効桁となるまで部分商を $n$ ビットづつくり返し求め, 商の最下位桁が求まった後, キャリーデータとサムデータを加算し, 剰余とする加算回路と, 剰余を被除数が桁合せした量だけ右シフトさせるシフトとを有していることを特徴とする固定小数点除算方式が得られる。

#### 〔実施例〕

次に, 本発明の実施例について図面を参照して説明する。

第1図は本発明の一実施例の除算回路である。

$n=2$ として説明する。1は除数, 被除数の有効桁の最上位桁を, 除数, 最初の剰余の最上位桁に桁合せする4進桁合せシフト, 2は除数を保持するレジスタ, 3はキャリーデータを保

#### 〔発明が解決しようとする課題〕

しかしながら従来の固定小数点除算方式は, 1サイクルの演算で商を1ビットづつ算出し, 除数, 被除数の桁数にかかわらず演算を実行しているため, 商と剰余を算出するまでに何サイクルも必要とした。また,  $n$ ビットづつ部分商を求める方式ではフルアダーを多数必要とし, ハード量が多くなる欠点がある。

そこで, 本発明の技術的課題は, 上記欠点に鑑み, 実行サイクルを低減した固定小数点除算方式を提供することである。

#### 〔課題を解決するための手段〕

本発明によれば, 除数, 被除数の有効桁の最上位桁を除数, 最初の剰余の最上位桁に桁合せする $2^n$ 進桁合せシフト( $n \geq 1$ )と, 桁合せされた除数を $+1, -1, -2, -4, \dots, -2^n$ 倍する倍数発生回路と, 除数, 被除数の有効桁より商の有効桁を事前に算出する有効桁算出回路と, 倍数発生回路の出力と桁合せされた剰余を入力し, 剰余から桁合せされた除数の仮の部分商倍

持するレジスタ, 4は被除数レジスタ, 5は除数, 被除数の有効桁より商の有効桁を算出する有効桁算出回路, 6は除数レジスタ, 7はキャリーデータとサムデータを加算し, 剰余を求めるアダー, 8は商の有効桁を保持するレジスタ, 9は桁合せされた除数を $+1, -1, -2, -4$ 倍する倍数発生回路, 10は剰余を被除数が桁合せした量だけ右シフトするシフト, 11~14は剰余と倍数発生回路の出力を減算するCBA, 12はCBAの出力の符号を求め負とならない最大の部分商を選択する選択回路である。以上が除算回路の構成要素である。

まず, 除数, 被除数は4進桁合せシフト1で桁合せされ, 桁合せされた除数はレジスタ2と除数レジスタ6へ格納され, 桁合せされた被除数は被除数レジスタ4へ格納される。除数, 被除数の有効桁より有効桁算出回路5で商の有効桁を算出し, レジスタ8へ格納する。除数レジスタ6の出力は倍数発生回路9へ入力され,  $-1$ 倍から $-3$ 倍までがCBA11~14で被

除数レジスタ4の上位桁(仮の剰余)と加算される。選択回路15では、CSA11~14の出力のキャリーデータとサムデータより仮の剰余の符号を求め、負とならない最大の仮の剰余および部分商を選択し、真の剰余と真の2ビットの部分商を得る。サムデータは被除数レジスタ4の上位桁へ格納され、下位桁は出力線50により2ビット左シフトされる。下位2ビットには、真の部分商が格納される。左シフトされた2ビットは被除数レジスタ上位桁の下位2ビットにシフトする。キャリーデータはレジスタ3へ格納され、次のサイクルでCSA11~14で加算される。CSA11~14は第2図の様に構成される。入力数53に除数レジスタ6の出力、51に被除数レジスタ4の出力の上位桁が入力される。 $n=2$ なので、倍数発生回路9の出力で、+1, +2, +3倍の除数をCSA11~14で演算する。その出力のキャリーデータとサムデータより仮の剰余の符号を求め、負とならない最大の仮の剰余および部分商を選

択し、真の剰余と真の2ビットの部分商を選択回路12で選択する。真の部分商、サムデータ、キャリーデータを出力線54, 55, 56より出力する。

以上の動作を有効桁だけくり返した後、レジスタ3に格納されたキャリーデータはサムデータとアダー7で加算され、レジスタ4の上位桁に格納される。このデータはシフタ10で被除数の桁合せ量だけ右シフトして剰余となる。商はレジスタ4の下位桁に格納される。

ここで、本方式を用いて除算を実例例で説明する。結果が出るまで5ステップに分けて第3図説明する。1ステップでは被除数、除数を00101110, 0110とする。このとき商の有効桁は2ケタと算出される。部分商を上位桁より $Q_2, Q_1$ とする。2ステップでは被除数、除数を4進桁合せシフタで桁合せする。その結果、被除数は10111000となり、除数は変わらない。3ステップでは $n=2$ なので被除数を上位2ビット拡張してから部分商 $Q_2$ を求める。

除数を-1, -2, -3倍し被除数の上位3桁(6ビット)と加算する。加算結果が負とならない最大の部分商 $Q_2$ は01である。4ステップでは部分商 $Q_1$ を求める。前ステップの加算結果1000101(部分商1)の下位2桁(4ビット)と、被除数0010111000の下位2桁(4ビット)と部分商1桁(2ビット)01を連結して、0101100001となる。この上位3桁(6ビット)と除数の-1, -2, -3倍を加算する。加算結果が負とならない最大の部分商 $Q_1$ は11である。この加算結果1000100(部分商11)の下位2桁(4ビット)と、被除数0101100001の下位2桁(4ビット)と、部分商11を連結すると0100000111となる。5ステップでは上位3桁(6ビット)を2ステップで桁合せした量だけ右シフトさせる。000100の下位2桁(4ビット)0100が剰余となり0100000111の下位2桁(4ビット)0111が商となる。

以下余白

#### [発明の効果]

以上説明したように本発明は、 $n$ ビットごとに商を求め、商の有効桁を事前に算出することにより、実行サイクルを減らす効果がある。また、CSAを用いることにより、ハード量を減らす効果がある。

#### 4. 図面の簡単な説明

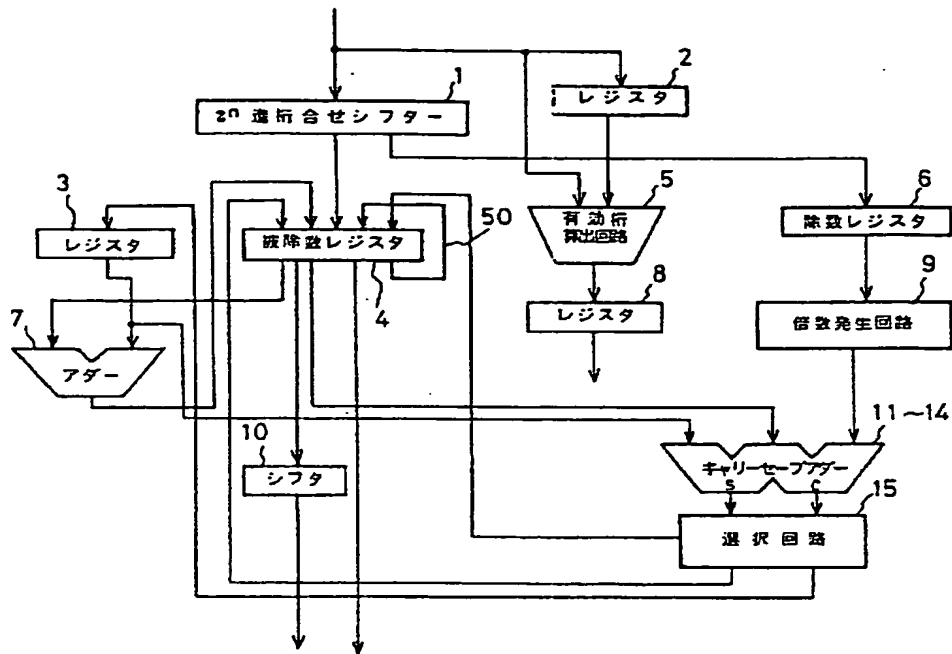
第1図は本発明の実施例で、第2図は倍数発生回路、CSA、選択回路の例を示した図で、第3図は除算の実例例で、第4図は従来例である。

1...2<sup>n</sup>進桁合せシフター、2, 3, 8, 16, 17, 18...レジスタ、4...被除数レジスタ、5...有効桁算出回路、6...除数レジスタ、7, 19...アダー、9...倍数発生回路、10...桁合せシフター、11~14...CSA、15...選択回路。

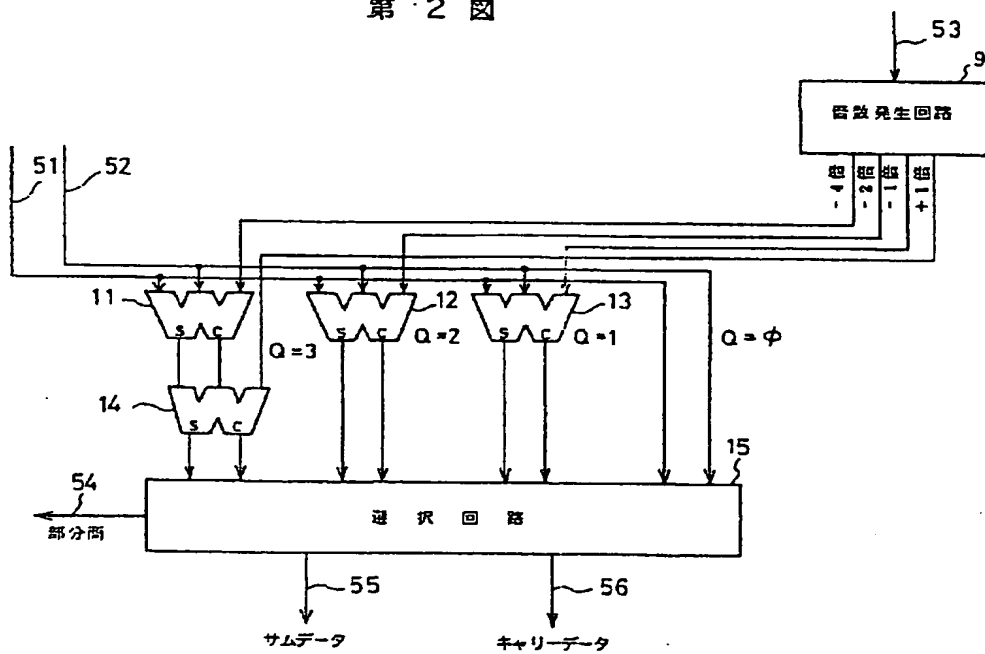
代理人 (7783) 弁理士 池田 憲 保



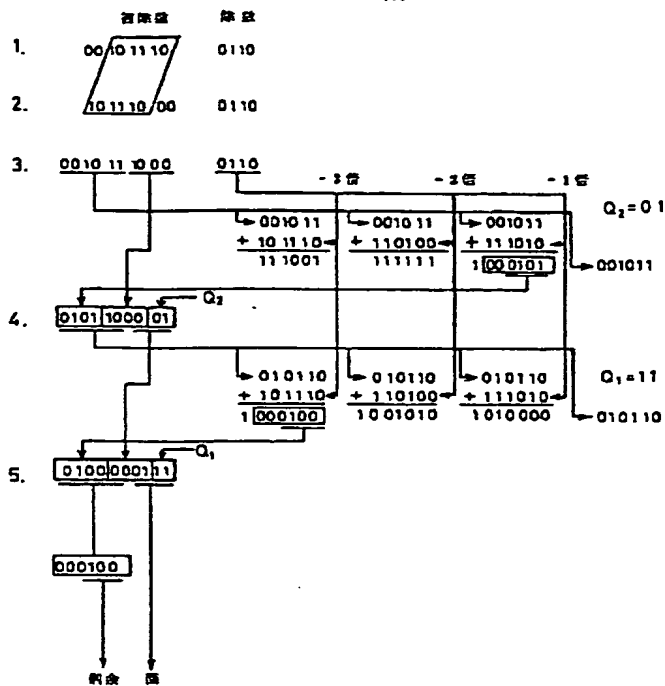
第 1 図



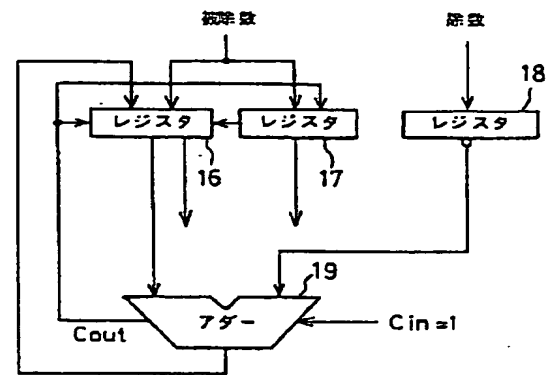
第 2 図



第 3 図



第 4 図



手続補正書 (自発)

平成 2 年 1 月 8 日

特許庁長官 吉田 文 毅 殿

1. 事件の表示

昭和 63 年特許願第 252905 号

2. 発明の名称

固定小数点除算方式

3. 補正をする者

事件との関係 特許出願人

名称 (423) 日本電気株式会社

名称 甲府日本電気株式会社

4. 代理人 〒105

住所 東京都港区西新橋 1 丁目 4 番 10 号

第三森ビル TEL (591)1507(代)

氏名 (7127) 弁理士 後 藤 洋 介

(ほか 2 名)

5. 補正の対象

①明細書全文

6. 補正の内容

方式 (明)

①別紙のとおり



明 細 書

1. 発明の名称

固定小数点除算方式

2. 特許請求の範囲

1) 正の整数除算を行う固定小数点除算方式において、

除数、被除数の有効桁の最上位桁を、桁合せする  $2^n$  進桁合せシフト ( $n \geq 1$ ) と、

該桁合せされた除数を  $+1$ 、 $-1$ 、 $-2$ 、 $-4$ 、 $\dots$ 、 $-2n$  倍する倍数発生回路と、

前記除数、被除数の有効桁より商の有効桁を事前算出する有効桁算出回路と、

前記倍数発生回路の出力と桁合せされた剰余を人力し、剰余から桁合せされた除数の仮の部分商倍 ( $+1$ 、 $+2$ 、 $+3$ 、 $\dots$ 、 $+(2^n - 1)$ ) された数をそれぞれ CSA トゥリーを用いて減算し、仮の剰余のキャリーデータとサムデータを得て、仮の剰余の符号を求め、負とにならない最小の仮の剰余

余および部分商を選択し、真の剰余と真の $n$ ビットの部分商を得る選択回路と、

商が有効桁となるまで部分商を $n$ ビットづつくり返し求め、商の最下位桁が求まった後、キャリーデータとサムデータを加算し、剰余とする加算回路と、

剰余を被除数が桁合せした量だけ右シフトさせるシフタとを有することを特徴とする固定小数点除算方式。

### 3. 発明の詳細な説明

#### 〔産業上の利用分野〕

本発明は固定小数点除算方式に関し、特に正の整数の除算で、商の有効桁を事前に求め、CSAを用いて部分商を $n$ ビット単位で有効桁だけ求める固定小数点除算方式に関する。

#### 〔従来の技術〕

正の整数除算を行う固定小数点除算方式は従来、商を1ビットづつ求めていた。この従来例を第4図に示す。被除数はレジスタ16、17へ上位、

で、商と剰余を算出するまでに何サイクルも必要とした。又、 $n$ ビットづつ部分商を求める方式ではフルアダーを多数必要とし、ハード量が多くなる欠点がある。

そこで、本発明の技術的課題は、上記欠点に鑑み、実行サイクルを低減した固定小数点除算方式を提供することである。

#### 〔課題を解決するための手段〕

本発明によれば、除数、被除数の有効桁の最上位桁を、桁合せする $2^*$ 進桁合せシフト( $n \geq 1$ )と、桁合せされた序数を $+1$ 、 $-1$ 、 $-2$ 、 $-4$ 、 $\dots$ 、 $-2^*$ 倍する倍数発生回路と、除数、被除数の有効桁より商の有効桁を事前に算出する有効桁算出回路と、倍数発生回路の出力と桁合せされた剰余を入力し、剰余から桁合せされた除数の仮の部分商倍( $+1$ 、 $+2$ 、 $+3$ 、 $\dots$ 、 $+(2^* - 1)$ )された数をそれぞれCSAトッリーを用いて減算し、仮の剰余のキャリーデータとサムデータを得て、仮の剰余の符号を求め、負とならない最小の仮の剰余および部分商を選択し、真の剰余と真の $n$ ビ

ットの下位と分けて格納され、除数はレジスタ18へ格納される。レジスタ16、17は連結してシフト動作する。レジスタ16に格納された部分被除数とレジスタ18より読み出力される除数は $Clr=1$ としてアダー19で減算が実行される。 $Cout=1$ ならば減算結果は正で、部分商は1となり、レジスタ17の下位1ビットに入力され、減算結果がレジスタ16へ入力される。 $Cout=1$ ならば、減算結果は負で、部分商は0となり、レジスタ17の下位1ビットに入力され、レジスタ16は前の値を保持する。減算を実行するごとにレジスタ17のデータは1ビットづつ左シフトされレジスタ16の下位1ビットへ入力される。

以上の動作をレジスタ17のデータがシフトし終わるまでくり返すことによりレジスタ16のデータが剰余にレジスタ17のデータが商になる。

#### 〔発明が解決しようとする課題〕

しかしながら従来の固定小数点除算方式は、1サイクルの演算で商を1ビットづつ算出し、除数、被除数の桁数にかかわらず演算を実行しているの

で、商と剰余を算出するまでに何サイクルも必要とした。又、 $n$ ビットづつ部分商を求める方式ではフルアダーを多数必要とし、ハード量が多くなる欠点がある。

#### 〔実施例〕

次に、本発明の実施例について図面を参照して説明する。

第1図は本発明の一実施例の除算回路である。

$n=2$ として説明する。1は除数、被除数の有効桁の最上位桁を、除数、最初の剰余の最上位桁に桁合せする4進桁合せシフト、2は除数を保持するレジスタ、3はキャリーデータを保持するレジスタ、4は被除数レジスタ、5は除数、被除数の有効桁より商の有効桁を算出する有効桁算出回路、6は除数レジスタ、7はキャリーデータとサムデータを加算し、剰余を求めるアダー、8は商の有効桁を保持するレジスタ、9は桁合せされた

除数を+1, -1, -2, -4倍する倍数発生回路, 10は剰余を被除数が桁合せした量だけ右シフトするシフタ, 11~14は剰余と倍数発生回路の出力を減算するCSA, 12はCSAの出力の符号を求め負とにならない最大の部分商を選択する選択回路である。以上が除算回路の構成要素である。

まず, 除数, 被除数は4進桁合せシフタ1で桁合せされ, 桁合せされた除数は除数レジスタ6へ格納され, 桁合せされた被除数は被除数レジスタ4へ格納される。除数, 被除数の有効桁より有効桁算出回路5で商の有効桁を算出し, レジスタ8へ格納する。除数レジスタ6の出力は倍数発生回路9へ入力され, -1倍から-3倍までがCSA 11~14で被除数レジスタ4の上位桁(仮の剰余)と加算される。選択回路15では, CSA 11~14の出力のキャリーデータとサムデータより仮の剰余の符号を求め, 負とにならない最大の仮の剰余および部分商を選択し, 真の剰余と真の2ビットの部分商を得る。サムデータは被除数レジ

スタ4の上位桁へ格納され, 下位桁は出力線50により2ビット左シフトされる。下位2ビットには, 真の部分商が格納される。左シフトされた2ビットは被除数レジスタ上位桁の下位2ビットにシフトする。キャリーデータはレジスタ3へ格納され, 次のサイクルでCSA 11~14で加算される。CSA 11~14は第2図の様に構成される。入力数53に除数レジスタ6の出力, 51に被除数レジスタ4の出力の上位桁が入力される。 $n=2$ なので, 倍数発生回路9の出力で, +1, +2, +3倍の除数をCSA 11~14で減算する。その出力のキャリーデータとサムデータより仮の剰余の符号を求め, 負とにならない最大の仮の剰余および部分商を選択し, 真の剰余と真の2ビットの部分商を選択回路15で選択する。真の部分商, サムデータ, キャリーデータを出力線54, 55, 56より出力する。

以上の動作を有効桁だけくり返した後, レジスタ3に格納されたキャリーデータはサムデータとアダー7で加算され, レジスタ4の上位桁に格納

される。このデータはシフタ10で被除数の桁合せ量だけ右シフトして剰余となる。商はレジスタ4の下位桁に格納される。

ここで, 本方式を用いて除算を実例で説明する。結果が出るまで5ステップに分けて第3図に説明する。1ステップでは被除数, 除数を00101110, 0110とする。このとき商の有効桁は2ケタと算出される。部分商を上位桁より $Q_2, Q_1$ とする。2ステップでは被除数, 除数を4進桁合せシフタで桁合せする。その結果, 被除数は10111000となり, 除数は変らない。3ステップでは $n=2$ なので被除数を上位2ビット拡張してから部分商 $Q_2$ を求める。除数を-1, -2, -3倍し被除数の上位3桁(6ビット)と加算する。加算結果が負とにならない最大の部分商 $Q_2$ は01である。4ステップでは部分商 $Q_1$ を求める。前ステップの加算結果1000101(部分商1)の下位2桁(4ビット)と, 被除数0010111000の下位2桁(4ビット)と部分商1桁(2ビット)01を連結して, 010

1100001となる。この上位3桁(6ビット)と除数の-1, -2, -3倍を加算する。加算結果が負とにならない最大の部分商 $Q_1$ は11である。この加算結果1000100(部分商11)の下位2桁(4ビット)と, 被除数0101100001の下位2桁(4ビット)と, 部分商11を連結すると0100000111となる。5ステップでは上位3桁(6ビット)を2ステップで桁合せした量だけ右シフトさせる。000100の下位2桁(4ビット)0100が剰余となり0100000111の下位2桁(4ビット)0111が商となる。

#### [発明の効果]

以上説明したように本発明は,  $n$ ビットごとに商を求め, 商の有効桁を事前に算出することにより, 実行サイクルを減らす効果がある。また, CSAを用いることにより, ハード量を減らす効果がある。

以下余白

4. 図面の簡単な説明

第1図は本発明の実施例で、第2図は倍数発生回路、CSA、選択回路の例を示した図で、第3図は除算の実施例で、第4図は従来例である。

1…2<sup>n</sup>進桁合せシフター、2、3、8、16、17、18…レジスタ、4…被除数レジスタ、5…付効桁算出回路、6…除数レジスタ、7、19…アダー、9…倍数発生回路、10…桁合せシフター、11～14…CSA、15…選択回路。

代理人 (7127) 弁護士 後藤 洋介

